**Online Judge System Specification**

By Jungle Wei

版权所有 不得复制

Copyright © Jungle Wei, weizengke.com

.  All Rights Reserved

1. 引言
   1. 编写目的和范围

本文主要介绍Online Judge平台系统的系统设计和使用方法，以指导后续开发和使用工作。本文预期读者为软件开发人员、测试人员以及Online Judge用户。

1. 软件概述
   1. 软件简述

Online Judge平台系统（简称OJ）是一个B/S架构的源程序判题系统。用户可以在线提交多种程序（如C、C++、Java）源代码，系统对源代码进行编译和执行，并通过预先设计的测试用例来检验程序源代码的正误。

用户可以在OJ系统上练习编程，参加竞赛，与其他用户讨论交流，提高自己的编程水平，并可以用于数据结构、程序设计教学的实验和考试。

当前最新版本版本：

[Judge Kernel 1.0.2](https://github.com/weizengke/JungleOS/releases/tag/1.0.2)

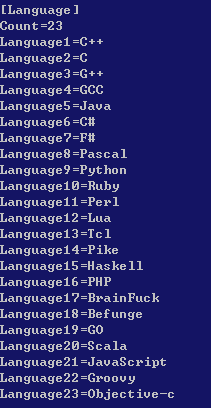
[Judge Web Platform 1.1.131215r4](https://github.com/weizengke/gdoj)

* 1. 支持特性

2.2.1Local Judge

Online Judge理论上支持所有可命令行编译和运行的程序语言；支持按语言分布式部署Jungle Kernel；支持动态添加Online Jungle平台程序语言；

当前已成功配置的程序语言如下：



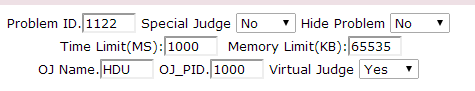
2.2.2Virtual Judge

当前版本支持两个OJ平台的Virtual Judge：

<http://acm.hdu.edu.cn> 以及<http://onlinejudge.guet.edu.cn/guetoj> 。

当前Virtual Judge实现主要使用Python语言实现爬虫，C语言适配Judge Kernel的架构方式。

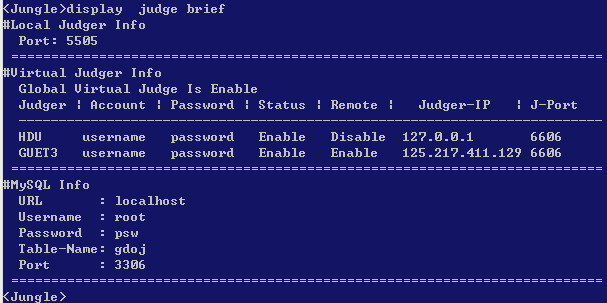
用户可以在Web后台将题目设置为Virtual Judge，并指明OJ平台及其题目实际编号。如下图：



***http://localhost/gdoj/admin/problemEdit.action?problemId=1122***

将本地题号为1122的题目设置为HDU平台的1000题，实现Virtual Judge。

查看当前judger详细配置：



当前支持virtual judger分布式部署：

命令如下：

全局使能去使能virtual judge：[undo] virtual-judge enable

使能去势能GUET-Vjudge：[undo] guet-judge enable

使能去使能分布式：[undo] guet-judge remote-judge enable

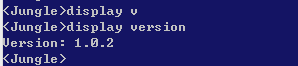
配置账号：guet-judge username STRING<1-24> password STRING<1-24>

配置分布式judger：guet-judge ip STRING<1-24> port INTEGER<1-65535>

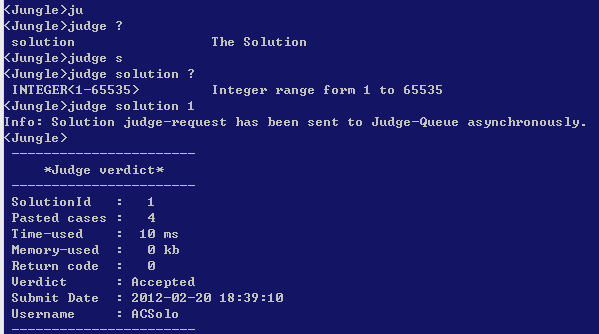
2.2.3Command Management

Online Judge Kernel支持命令行管理，命令行特性支持命令联想、自动补全等功能，极大方便了系统的管理。

查看当前版本号的命令如下：



重判Solution ID为1的提交的命令如下：



* 1. 运行环境

操作系统：Windows XP/7/8/8.1 Windows Server 2003/2008

JDK：Java JDK1.5以上；

Web容器：Apache Tomcat 5.5以上；

数据库：MySQL 5.1以上；

编译环境：GCC、JAVA、Python、Ruby等编程语言编译环境。

1. 系统安装
   1. 安装配置JDK

OJ系统对JDK安装位置没有要求，直接安装完成。以安装C:\Program Files\Java\jdk1.6.0\_21为例。

右击 我的电脑，找到 属性-> 高级-> 环境变量-> 系统变量，添加以下环境变量：

JAVA\_HOME=C:\Program Files\Java\jdk1.6.0\_21

path=%JAVA\_HOME%\bin

* 1. 安装配置Tomcat

OJ系统对Tomcat安装位置没有要求，直接安装完成。以安装D:\tomcat为例。

如需修改端口，可修改D:\tomcat\conf目录下的server.xml文件找到Connector port=“8080”，将8080端口改为其他端口（如80），这里以修改为80端口为例。

启动Tomcat服务，在浏览器中输入http://127.0.0.1:80/, 若出现Tomcat的默认页面，则安装成功。

* 1. 安装配置MySQL

MySQL安装路径没有要求，这里以D:\MySQL为例，字符集选择GBK，端口选择默认的3306，设置用户名root，密码rootpwd。

修改D:\MySQL\my.ini文件，在最后添加wait\_timeout=2880000。

重启MySQL服务。

导入数据库文件gdoj.sql，以创建OJ系统的数据库。

* 1. 安装配置OJ系统判题内核程序

将OJ文件夹拷贝到判题服务器上（请勿随意移动文件夹内的文件），以D:\目录为例。

修改D:\OJ\conf\config.ini文件MySQL配置如下：

[MySQL]

url=127.0.0.1

username=root

password=rootpwd

table=gdoj

port=3306

其中url为数据库服务器ip，这里以与判题服务器使用同一服务器为例，username为数据库用户名，password为数据库密码，table为数据库名，port为数据库端口。

[Tool]

DeleteTemp=0

WorkingPath=OJ\_TMP\

DataPath=data\

JudgePath=Judge.exe

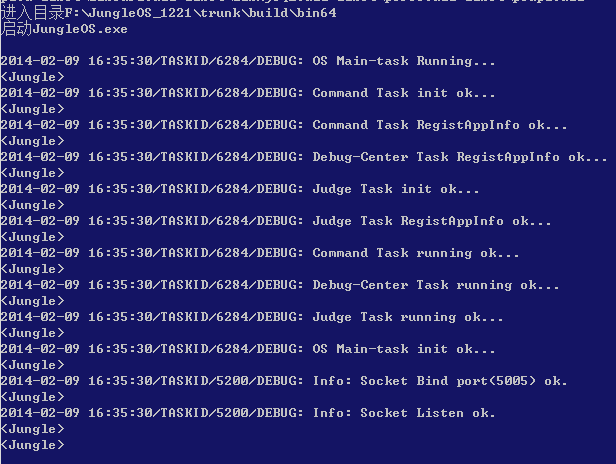
LimitJudge=2000000

Port=5005

OutputLimit=1048576

其中DeleteTemp为是否删除临时文件（0：不删除，1删除），WorkingPath为临时文件产生目录，DataPath为测试数据目录，LimitJudge为最大允许的裁判等待数量，Port为Socket通信端口，OutputLimit为最大允许用户输出的字节数。

启动判题内核程序Judge.exe，若看到如下界面则为启动成功。



判题内核启动成功界面

* 1. 安装配置OJ系统Web网站

将gdoj.war包发布到Tomcat中，直接拷到D:\Tomcat\webapps\下即可。

**修改D:\Tomcat\webapps\gdoj\WEB-INF\classes\ config.properties文件如 下：**

**OJ\_PATH=D\:\\OJ\\**

**OJ\_DATA\_PATH=D\:\\OJ\\data\\**

**OJ\_TMP=D\:\\OJ\\OJ\_TMP**

**OJ\_INI\_PATH=D\:\\OJ\\conf\\data.ini**

**其中OJ\_PATH为OJ文件夹绝对路径，OJ\_DATA\_PATH为测试数据绝对路径，OJ\_TMP为临时文件绝对路径，OJ\_INI\_PATH为配置文件config.ini绝对路径。**

修改D:\Tomcat\webapps\gdoj\WEB-INF\classes目录下的hibernate.cfg.xml和 applicationContext.xml文件中的数据库url为数据库服务器url。

重启Tomcat服务，打开浏览器访问http://127.0.0.1:8080/gdoj/，若出现OJ系统网站首页，则说明安装成功。

* 1. 如何部署多台判题服务器

目前，OJ系统支持的分布式判题模式为：将不同语言的判题部署到不同的服务器。

具体操作如下（这里以添加C++为例）：

1. 打开config.ini配置文件，在[Language]字段中添加一下内容（格式Language+N=语言名称，其中N为第几种语言）：

Language1=C++

1. 在[Transcoding]字段添加C++语言是否需要字符转换，0为不需要，1为需要，假设C++编译环境为VS2010，则需要转换，所以添加以下内容：

C++=1

1. 在[TimeLimit]字段添加C++语言需要的时间、内存限制倍数，这里不需要增倍，所以添加如下内容：

C++=1

1. 在[ProcessLimit]字段添加C++语言在的系统运行过程中允许的进程个数，这里只允许一个，所以添加如下内容：

C++=1

1. 在[LanguageExt]字段添加C++语言的源代码后缀名，这里是cpp，所以添加如下内容：

C++=cpp

1. 在[LanguageExe]字段添加C++语言可执行文件的后缀名，这里为exe，所以添加如下内容：

C++=exe

1. **在[JudgeIP]字段添加C++判题服务器的IP地址**，这里为假设为125.217.41.211，所以添加如下内容：

C++=125.217.41.211

1. 在[SourcePath]字段添加C++语言源代码程序的路径，这里添加如下内容：

C++=%PATH%%NAME%.%EXT%

1. 在[ExePath]字段添加C++语言可执行程序的路径，这里添加如下内容：

C++=%PATH%%NAME%.%EXT%

1. 在[CompileCmd]字段添加C++语言的编译命令，这里添加如下内容：

C++=cl /TP /F16777216 /EHsc /O2 /DONLINE\_JUDGE %PATH%%NAME%.%EXT% /Fo%PATH%%NAME% /Fe%PATH%%NAME% 1>%PATH%%NAME%.txt

1. 在[RunCmd]字段添加C++语言的运行命令，这里添加如下内容：

C++=%PATH%%NAME%.%EXE%

注意：若所添加的语言是脚本语言，不许编译，则将[CompileCmd]中的值设为NULL。

然后打开Language.xml文件，添加C++语言的信息，这里的信息将会显示在提交页面的语言选择菜单中。这里写入：

<VALUE>

<ID>1</ID>

<LANG>Microsoft Visual Stdio 2010 C++</LANG>

</VALUE>

修改D:\Tomcat\webapps\gdoj\WEB-INF\classes\globalMessages.properties文件，这里是显示在提交状态的语言名称。

languageCount=1

language0=All

language1=MS C++

**最后重新发布gdoj项目即可**。

* 1. 配置Virtual Judge账号

Config.ini配置文件种的[HDU]字段为HDU Virtual Judge相关配置，当前只需要配置在HDU上的机器人账号和密码

username=hduhandle

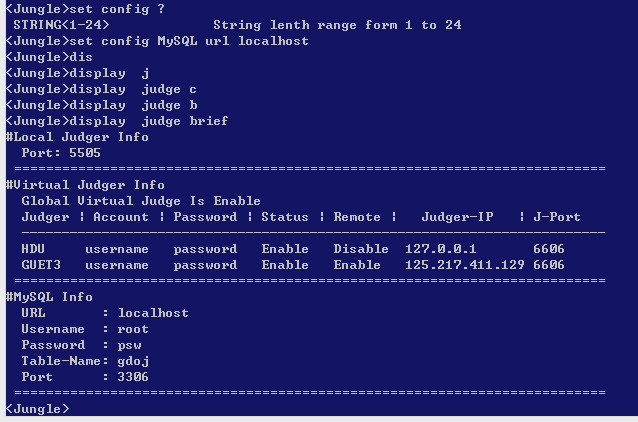
password=hdupsw

* 1. 可配置管理

Set Config section name value:

**set config STRING<1-24> STRING<1-24> STRING<1-65535>**

注意：该命令行需要重启生效。



1. More

更多功能有待挖掘，这里无法详细说明。

更多帮助？请联系Jungle，可在<http://weizengke.com>留言。